

UNIVERSITY OF TECHNOLOGY, SYDNEY

**Efficient Algorithms and a Two-Stage
Framework for Autonomous Exploration
of Complex 3D Environments Using a
Climbing Robot**

by

Phillip Quin

A thesis submitted in partial fulfillment for the
degree of Doctor of Philosophy

in the

Faculty of Engineering and IT
Intelligent Mechatronic Systems Group

October 2016

Declaration of Authorship

I, Phillip Quin , declare that this thesis titled, ‘Efficient Algorithms and a Two-Stage Framework for Autonomous Exploration of Complex 3D Environments Using a Climbing Robot’ and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Date:

UNIVERSITY OF TECHNOLOGY, SYDNEY

Abstract

Faculty of Engineering and IT
Intelligent Mechatronic Systems Group

Doctor of Philosophy

by Phillip Quin

Enabling robots to autonomously explore complex 3D environments is crucial in facilitating the automation of many real-world tasks. There exist many algorithms for exploring unknown environments with autonomous robots. Most of these are restricted to the 2D case, or to cases where the robot can be abstracted as a holonomic point robot. Algorithms that deal with the 3D case restrict the robot’s possible positions to the 2D plane, or assume that the robot can freely move through any empty space, like an idealised quadrocopter.

This thesis presents a two-stage exploration framework that allows robots to consider any adherable surface in a 3D environment as a potential position from which to conduct exploration. The framework is therefore suitable to any robotic platform that must at all times maintain contact with a surface, but where this surface need not be the floor plane.

A Nearest Neighbours Exploration Approach (NNEA) is developed to accomplish exploration of the environment immediately surrounding the robot when the robot is fixed to a position on a surface. In this approach, the Next Best Viewpoint is selected first by evaluating and choosing between candidate viewpoints that are within a bounded range of the robot’s current position. NNEA is demonstrated in experiments in a real bridge environment for the case of a high degrees of freedom (DOF) robot arm with a fixed base. NNEA is shown to result in faster exploration times in the case of a high-DOF robot arm in a fixed base position.

Four frontier detection algorithms are proposed and investigated for determining the set of frontiers—the boundary between known and unknown space—after each map update. The resulting frontiers are used to limit which candidate positions need to be considered for exploration. The novel frontier detection algorithms are compared to other state of the art algorithms and are found to be suited for efficient frontier detection in different situations.

A novel graph-based method for selecting the Next Best Base location (NBB) is presented in which the map is used to create an updated graph of possible positions for the robot base, sampled from all surfaces. Positions that are sufficiently close to the frontiers are selected as candidate positions for the robot to move to next. The information that could be gained from each reachable candidate position is estimated. A cost function determines which candidate is the best to move to next, and the robot moves to that position to take

another sequence of scans. This method is demonstrated in simulations and experiments to be efficient in minimising the computation required to select and move to the NBB.

The exploration framework and the developed algorithms and approach are demonstrated in simulation in an environment made up of unconnected surfaces, large enough that the robot is required to repeatedly move through the environment in order to fully explore it. The framework is shown to result in efficient exploration of the observable environment.

Acknowledgements

If you want to build a ship, don't drum up people together to collect wood and don't assign them tasks and work, but rather teach them to long for the endless immensity of the sea.

— Antoine de Saint-Exupéry

I would like to thank my supervisor Professor Dikai Liu for providing guidance and support, particularly when it came to the bigger picture of the thesis, and for making it possible to be a part of the Bridge Climbing Robot project through the ARC linkage grant and in collaboration with the Roads and Maritime Services NSW.

I am also grateful to all the staff and students at the Centre for Autonomous Systems (CAS) for providing a warm and supportive environment for research. The Bridge Climbing Robot project of which my thesis is a part would not have been as interesting, entertaining or feasible without the participation of many others, including Peter Ward, David Pagano, Chia-Han “John” Yang, and Liyang Liu.

I owe a lot to my co-supervisors Dr. Alen Alempijevic and Dr. Gavin Paul. Gavin gave me the tools to get started, and both spent many hours reading and giving feedback on my work, papers, and presentations. If ever I was stuck or at a mental dead-end, I could be sure that explaining my train of thought to Alen or Gavin, and our subsequent discussions, would result in some new insight. When I grow up, I want to be as helpful and reliable as they are to those around them. Хвала and 再度感謝します to both of you for being my 地獄に仏 when био сам млад и океан било дубоко.

My parents instilled in me a love of science, engineering, and discovery. More dubiously, they gave me the confidence to take up the offer of a PhD scholarship in the first place. *Mais petit a petit, l'oiseau fait son nid*, and they followed through by giving me the support and confidence to see the thesis through to the end, otherwise I might never have forgiven them.

Thanks go to my brother Adam, for being fun to hang out with, when that was exactly what I needed, and to my brother Eric, for making me want to become a doctor before he did (I lost); even if we might argue about what a “real” doctor is.

To all my friends I am indebted for helping me maintain some modicum of stability and perspective when I otherwise would have been overwhelmed. Thanks for being fun people to spend time with.

To Glenn Shea, Carrie Browne, and Mitesh Patel, for having been-there-done-that, Helen Fearnley for being a trailblazer, and Maximilian Wittmann for proving that even PhDs come to an end. Particular thanks to Denny Krstevski, for keeping me engaged in artistic pursuits. To Tom Hsu, 感谢您提供住房和酒精.

To Andrija Krstic, for being so large in life and in the memories of his loved ones that he is able to inspire someone who will never have the pleasure of meeting him. From him I hope I've at least learnt што можеш данас, не остављај за сутра.

Finally, Nevenka Krstic, моје светло у олуји. This thesis would not have happened without you.

Contents

Declaration of Authorship	iii
Abstract	iv
Acknowledgements	vii
List of Figures	xiii
List of Tables	xvii
Abbreviations	xix
Nomenclature	xxi
Glossary of Terms	xxv
1 Introduction	1
1.1 Background and Motivation	4
1.2 Research Questions	8
1.3 Scope	9
1.4 Contributions	10
1.5 Publications	11
1.5.1 Directly Related Publications	11
1.5.2 Related Publications	12
1.6 Thesis Outline	12
2 Review of Related Work	15
2.1 Mapping and Localisation in Complex 3D Environments	16
2.1.1 Metric Representations	16
2.1.2 Topological Representations	20
2.1.3 Simultaneous Localisation and Mapping (SLAM)	21
2.2 Path Planning and Collision Avoidance	22
2.3 Exploration: Semi-Autonomous vs. Autonomous	25
2.4 Exploration: Targets, Known Environments, and Undirected Strategies	27

2.4.1	Searching for a Target in Known Environments	27
2.4.2	General Exploration in Known Environments	27
2.4.3	Undirected Exploration	29
2.5	Exploration: Directed Investigation of Unknown Space	31
2.5.1	Wall-Following	32
2.5.2	Exploration as Path Planning By-Product	32
2.5.3	Next Best View Exploration	33
2.5.4	Utility and Frontier Based Exploration	34
2.5.5	Graph-Based Exploration	38
2.5.6	C-space Based Exploration	40
2.5.7	Exploration Termination Criteria	42
2.6	Summary	43
3	Two-Stage Exploration Framework	47
3.1	Description	49
3.1.1	Local Exploration	49
3.1.2	Determining and Moving to the Next Best Base Location	51
3.2	Conclusions	52
4	Nearest Neighbour Exploration Approach	53
4.1	Overview	53
4.2	Nearest Neighbour Exploration Approach	56
4.2.1	Finding Nearest Neighbour NBV	56
4.2.2	Sampling all C-space for NBV	60
4.2.3	Nearest Neighbour Exploration Approach with Backtracking	62
4.3	Complexity of Calculations	65
4.4	Experiments and Simulations	70
4.4.1	Robot and Environment	70
4.4.2	Experiment Design	72
4.4.3	Results	74
4.5	Engineering Challenges	80
4.5.1	Sensor Minimum Range	80
4.5.2	Managing Occlusion of Sensor FOV by the Robot	81
4.5.3	Deflection of Robot Joints Due to Gravity	82
4.6	Discussion	82
4.6.1	Quality of Environments Used	83
4.6.2	Estimation of Possible Information Gain	83
4.6.3	Analysis of Maximal Effort as a Predictor of Time Spent in Motion	86
4.6.4	Choosing Information Thresholds	88
4.7	Conclusions	88
5	Frontier Detection Algorithms	91
5.1	Nomenclature	92
5.2	Naïve Active Area Frontier Detection	93
5.2.1	Description	93

5.2.2	Soundness and Completeness	94
5.2.3	Analysis	94
5.3	Expanding Wavefront Frontier Detection	95
5.3.1	Description	96
5.3.2	Soundness and Completeness	97
5.3.3	Algorithm Analysis	98
5.4	Frontier-Tracing Frontier Detection	101
5.4.1	Description	103
5.4.2	Soundness And Completeness	105
5.4.3	Analysis	106
5.5	State-Based Frontier Detection	108
5.5.1	Analysis	112
5.6	Experiments	113
5.6.1	Design	113
5.6.2	Results	114
5.7	Discussion	121
5.7.1	Summary of Complexity	121
5.7.2	Choosing a Frontier Detection Approach	123
5.8	Conclusions	124
6	A Graph-Based Method for Selecting the Next Best Base Location	127
6.1	Description	127
6.1.1	Determining Attachment Points	129
6.1.2	Estimating Candidate Information Score	132
6.1.3	Selecting and Moving to the Next Best Base Location	132
6.2	Analysis	136
6.3	Experiments	139
6.4	Discussion	141
7	Case Study: Exploration of Multi-Surface Environments	147
7.1	Experiment Design	148
7.1.1	The Environment and the Robot	148
7.1.2	System Implementation	149
7.1.3	Evaluation Criteria	150
7.2	Results	152
7.3	Discussion	156
7.3.1	The Utility Function	156
7.3.2	Execution Time	158
7.3.3	m-A*, Connectivity, and Edge Validity	162
8	Conclusions	163
8.1	Summary of Contributions	164
8.1.1	A Two-Stage Exploration Framework for Climbing Robots	164
8.1.2	A Novel Manipulator-Based Exploration Approach	164
8.1.3	Four Novel Frontier Detection Algorithms	165

8.1.4	A Graph-Based Method for Selecting the Next Best Base Location .	165
8.1.5	Practical Contribution	166
8.2	Discussion of Limitations	166
8.3	Future Work	167
Appendices		169
A	NNEA Experimental Data	171
A.1	Exploration Metrics	171
A.2	Time Profiles	174
B	Existing Frontier Detection Algorithms	177
B.1	Naïve Frontier Detection	178
B.2	Wavefront Frontier Detection	178
B.3	Fast Frontier Detection	179
B.4	Incremental Wavefront Frontier Detection	182
C	Frontier Detection Experimental Data	185
C.1	Environments and Trajectories	185
C.2	Total Time and Operations	187
C.3	Computation Per Iteration Over Time	189
C.4	Trend Analysis	192
C.5	Tables	192
Bibliography		197

List of Figures

1.1	Skycleaner 3, Hulltimo, and Expliner.	4
1.2	Bridge inspectors in Canada. (Credit: CRAS)	6
1.3	Bridge inspectors in Missouri, USA. (Credit: MoDOT, KBIA)	6
1.4	The caterpillar-inspired 7DOF robot platform.	7
2.1	An occupancy grid.	17
2.2	A laser return being incorporated into an occupancy grid.	18
2.3	A multi-level surface map.	18
2.4	A quadtree structure.	19
2.5	A floorplan and a topological map.	21
2.6	A Voronoi graph built from an environment.	21
2.7	Ellipsoids used to detect and prevent collisions.	25
2.8	Camera placements in an art gallery [1].	28
2.9	Asymptotically optimal coverage planner.	29
2.10	Demonstration that exploration of unknown environments can only be greedy.	30
2.11	A Levy walk.	31
2.12	Following walls to travel through the environment.	32
2.13	Exploration as a by-product of path planning.	33
2.14	Frontier cells.	36
2.15	A potential field and harmonic function.	37
2.16	Exploration using SRT-Ball and SRT-Star.	39
2.17	Reducing C-space entropy.	41
2.18	Thresholds for ending exploration.	42
2.19	Getting stuck when moving the sensor towards frontiers.	44
2.20	Regions missed when only considering ground plane.	45
3.1	The two-stage exploration framework.	48
3.2	Example configurations from a local exploration strategy.	50
3.3	Attachment points as nodes in a graph.	51
4.1	A sequence of greedy optimal observations.	55
4.2	The robot “painting itself into a corner”.	56
4.3	The Nearest Neighbours Exploration Approach.	57
4.4	Choosing $\Delta\theta$ based on the discretisation of \mathcal{P}	59
4.5	Nearest Neighbours Exploration Approach with Backtracking.	63
4.6	The tree T of poses generated by NNEA-B.	64

4.7	The number of evaluations required by an optimal planner and NNEA. . . .	66
4.8	The percentage of calculations saved by NNEA.	68
4.9	Values of m and k for which NNEA and NNEA-B are worthwhile.	69
4.10	The robot in the simulated environments.	71
4.11	The robot in the lab and bridge environments.	72
4.12	Performance in simulated environments.	74
4.13	Performance in lab and bridge environments.	75
4.14	Meshes resulting from exploration.	76
4.15	Time taken by each task in the lab and bridge environments.	77
4.16	Information over time.	79
4.17	Joint deflection and managing unknown cells within minimum range.	81
4.18	Filling in free space when receiving 0 range reading.	81
4.19	(a) Bridge site 1 and (b) bridge site 2 seen from the side.	83
4.20	Estimating information as a function of the number of sensor observations.	85
4.21	Estimating the maximum observable information as a function of $ Q_a $	86
4.22	Maximal effort vs. movement time.	87
4.23	Choosing between Local Exploration algorithms.	89
5.1	Cells in $\mathcal{P}_{free}^t - \mathcal{P}_{free}^{t-1}$ which will be evaluated at time t	98
5.2	The cells evaluated by FTFD and EWFD.	102
5.3	Relationships between the sensor FOV and frontiers.	102
5.4	An example of how a pocket could form.	103
5.5	A cell/voxel and its neighbours in an occupancy grid.	110
5.6	The Cave and Freiburg simulated environments.	113
5.7	Comparison of each algorithm's running time.	116
5.8	Computation time at each iteration in Freiburg trajectory H.	117
5.9	Computation time of frontier detection vs. number of known freespace cells.	118
5.10	Preliminary experiments using real 2D data.	120
6.1	A flowchart of processes involved in the exploration strategy.	128
6.2	Example attachment positions.	129
6.3	Two attachment positions.	130
6.4	Attachment nodes and the edges between them.	131
6.5	Methods for estimating information gain.	133
6.6	The robot moving through a simple plank environment.	140
6.7	The robot in the beams environment, consisting of three disjoint surfaces.	142
6.8	The robot in the box environment, consisting of four surfaces.	143
6.9	The robot in the lab rig.	144
6.10	Time taken by each task when determining the NBB.	145
7.1	A wireframe of the simulated environment.	152
7.2	Visualisations of the robot exploring the environment.	153
7.3	Visualisations of the robot exploring the environment.	154
7.4	Portion of estimated information remaining over time.	155
7.5	The estimated information of evaluated nodes at each iteration.	157

7.6	The estimated effort of evaluated nodes at each iteration.	157
7.7	The estimated utility of evaluated nodes at each iteration.	158
7.8	Time taken by each task.	159
7.9	Detailed time taken by each task.	160
7.10	Time taken by each individual task.	161
B.1	Rays traversing the occupancy grid.	182
C.1	The Freiburg environment trajectories G (red), H (green), I (blue).	186
C.2	The Freiburg environment trajectories J (red), K (green), L (blue).	186
C.3	The Freiburg environment trajectories M (red), N (green), O (blue).	186
C.4	Cave trajectory summary (A, B, C).	187
C.5	Freiburg trajectory summary (D, E, F).	187
C.6	Freiburg trajectory summary (G, H, I).	188
C.7	Freiburg trajectory summary (J, K, L).	188
C.8	Freiburg trajectory summary (M, N, O).	188
C.9	Computation time at each iteration in Cave trajectory A.	189
C.10	Computation time at each iteration.	190
C.11	Computation time at each iteration.	191
C.12	Computation time at each iteration.	192

List of Tables

4.1	Percentages of ground truth collected in each bridge site.	83
5.1	Example time taken for Naïve frontier detection.	92
5.2	Correlation between the time taken to compute frontiers and...	117
5.3	Comparing computation time in seconds per 1000 cells for SBFD and CBFD.	119
5.4	Minimum sizes of \mathcal{F}_t for $ A_t < \mathcal{F}_{t-1} ^{(d-1)/d}$ to be true.	123
5.5	Order of complexity of each frontier detection algorithm.	123
5.6	Order of complexity of each algorithm without frontier grouping.	123
5.7	Assumptions and properties of frontier detection algorithms.	125
6.1	Time taken by each task.	141
6.2	Size of graph and number of nodes and edges evaluated.	141
7.1	Experiment parameters.	151
A.1	Results for Exploration in lab site 1.	171
A.2	Results for Exploration in lab site 2.	172
A.3	Results for Exploration in lab site 3.	172
A.4	Results for Exploration in lab site 4.	172
A.5	Results for Exploration in lab site 5.	173
A.6	Results for Exploration in bridge site 1.	173
A.7	Results for Exploration in bridge site 2.	173
A.8	Seconds spent performing each subtask during exploration in lab site 1. . .	174
A.9	Seconds spent performing each subtask during exploration in lab site 2. . .	174
A.10	Time (s) spent performing each subtask during exploration in lab site 3. . .	175
A.11	Time (s) spent performing each subtask during exploration in lab site 4. . .	175
A.12	Time (s) spent performing each subtask during exploration in lab site 5. . .	175
A.13	Time (s) spent performing each subtask during exploration in bridge site 1.	176
A.14	Time (s) spent performing each subtask during exploration in bridge site 2.	176
C.1	Correlation between the time taken to compute frontiers and...	192
C.2	Results for Cave trajectory A.	193
C.3	Results for Cave trajectory B.	193
C.4	Results for Cave trajectory C.	193
C.5	Results for Freiburg trajectory D.	193
C.6	Results for Freiburg trajectory E.	193

C.7	Results for Freiburg trajectory F.	193
C.8	Results for Freiburg trajectory G.	193
C.9	Results for Freiburg trajectory H.	194
C.10	Results for Freiburg trajectory I.	194
C.11	Results for Freiburg trajectory J.	194
C.12	Results for Freiburg trajectory K.	194
C.13	Results for Freiburg trajectory L.	194
C.14	Results for Freiburg trajectory M.	194
C.15	Results for Freiburg trajectory N.	194
C.16	Results for Freiburg trajectory O.	195

Abbreviations

2D	2 Dimensional
3D	3 Dimensional
AXBAM	Autonomous eXploration to Build A Map, system
BFS	Breadth-First Search
C-space	Configuration Space of manipulator
CBFD	Change-Based Frontier Detection
DFS	Depth-First Search
DOF	Degrees of Freedom
EWFD	Expanding Wavefront Frontier Detection
FFD	Fast Frontier Detection
FOV	Field Of View
FTFD	Frontier-Tracing Frontier Detection
NBB	Next Best Base location
NBV	Next Best Viewpoint
NN	Nearest Neighbour
NNEA	Nearest Neighbour Exploration Approach
NNEA-B	Nearest Neighbour Exploration Approach with Backtracking
RGB-D	Red Green Blue Depth camera.
RRT	Rapidly-exploring Random Trees
SBFD	State-Based Frontier Detection
SLAM	Simultaneous Localisation And Mapping
WFD	Wavefront Frontier Detection
WFD-INC	Incremental Wavefront Frontier Detection

Nomenclature

General Formatting Style

$f(\dots)$	A scalar valued function
$\mathbf{f}(\dots)$	A vector valued function
$[\dots]^T$	Transpose
$ \cdot $	Absolute value
$\ \cdot\ $	Vector length and normalised vector
$A \setminus B$	The set A without any elements that are also in B. Equivalent to $B^c \cap A$.
\mathbb{N}^0	Natural numbers greater or equal to zero; e.g. 0, 1, 2, 3, ...
$O(g(x))$	Big O notation. $f(x) = O(g(x))$ means $f(x)$ is bounded above asymptotically by $g(x)$. I.e. $ f(x) \leq k \times g(x) $, for sufficiently large x , and some positive k .
$\Omega(g(x))$	Big Omega notation. $f(x) = \Omega(g(x))$ means $f(x)$ is bounded below asymptotically by $g(x)$. I.e. $ f(x) \geq k \times g(x) $, for sufficiently large x , and some positive k .
\mathbb{R}^2	The set of points in 2D Euclidean space; e.g. (1.2, -57.5).
\mathbb{R}^3	The set of points in 3D Euclidean space; e.g. (-3.4, 105.19, 8).
$\Theta(g(x))$	Big Theta notation. $f(x) = \Theta(g(x))$ means $f(x)$ is bounded asymptotically above and below by $g(x)$. I.e. $k_1 \times g(x) \leq f(x) \leq k_2 \times g(x) $, for sufficiently large x , and some positive k_1 and k_2 .

Nearest Neighbour Exploration Approach

\mathcal{A}	The robot.
---------------	------------

$\mathcal{A}(\mathbf{q})$	The space taken up by the robot in configuration \mathbf{q} .
\mathcal{C}	Configuration space.
$\Delta\theta$	The angle used to determine neighbouring configurations.
$\mathcal{H}(\mathbf{q})$	The information value of a configuration \mathbf{q} .
\mathcal{P}_{free}^t	The set of known free-space at time t .
\mathcal{P}_{obs}^t	The set of known obstacles at time t .
\mathcal{P}_{unk}^t	The set of unknown space at time t .
Q_a	The set of candidate configurations under consideration by AXBAM.
Q_n^t	The set of candidate configurations under consideration by NNEA at time t .
\mathbf{q}	A robot configuration, representing a vector of joint angles.
\mathbf{q}_{curr}^t	The robot configuration at time t .
\mathbf{q}_{nbv}^t	The next best view configuration at time t .
t	A variable representing time.
${}^oT_s^{\mathbf{q}}$	The transform from map origin to sensor origin based on the robot configuration \mathbf{q} .
τ_a	The information threshold under which configurations will be discarded in the AXBAM algorithm.
τ_n	The information threshold under which configurations will be discarded in the Nearest Neighbours phase of NNEA.

Frontier Detection Algorithms

A_t	The active area of the observation at time t .
A_{max}	The largest size of any possible active area.
\mathcal{E}_{max}	The largest size that any set of cells forming the boundary of any $\mathcal{S}(O_t)$ could be.
$\mathcal{E}(O_t)$	The set of cells in $\mathcal{S}(O_t)$ that are on the edge of the map or adjacent to cells not in $\mathcal{S}(O_t)$.
\mathcal{F}_{aa}	The set of known frontiers inside the active area under consideration.
\mathcal{F}_{new}	The set of known frontiers that, when added to the set F_{t-1} , will make it become F_t .

\mathcal{F}_t	The set of known frontiers at time t .
\mathcal{F}_{t-1}	The set of known frontiers at time $t - 1$.
M	The set of cells in the map.
M_{free}	The set of free-space cells in the map.
O_t	The sensor observation made at time t .
o_t	The number of individual observations made as part of O_t .
\mathcal{S}_{max}	The largest size that any set of cells covered by the sensor could be.
$\mathcal{S}(O_t)$	The set of cells covered by the observation O_t .

Selecting the Next Best Base Location

A	The set of valid attachment positions.
\mathcal{C}	The set of candidate attachment positions.
$C(a, a_0)$	The cost of reaching attachment position a given the robot begins at a_0 .
d_{max}	The maximum transition step the robot can make across a surface.
d_{min}	The minimum transition step the robot can make across a surface.
$e_{i,j}$	An edge linking attachment position i and j .
\mathcal{F}	The set of frontiers.
$H(a)$	A function returning the estimated information value of an attachment position a .
\mathcal{M}_o	The map of free and occupied space.
\mathcal{M}_s	The map of known surfaces.
$\mathbf{p}_{i,j,k}$	A robot trajectory assuming the robot starts with the end effector at to attachment position i , the base affixed to attachment position j , and the end effector will be moved to attachment position k .
$\mathbf{q}_{j,k}$	A robot configuration “straddling” the base positions j and k .
r_l	The length of the robot.
r_s	The trustworthy range of the sensor.
$s()$	The sampling policy which takes as argument a map of surfaces.
0T_b	The homogeneous transform describing the robot base.

τ_s	The threshold under which an attachment position is considered unsafe.
$U(g, s)$	The utility function taking as arguments a goal node g and start node s .
V_r	A sphere of radius r .

Glossary of Terms

Autonomous	Without human intervention.
Base	The footpad of the robot that is currently affixed to the surface.
Binary Tree	A tree data structure in which each node has at most two children.
Bounding Box	The smallest box, usually axis-aligned, that can be made to contain a set of given points.
Candidate	A node/position considered as the next potential position from which the robot should make new observation(s) of the environment.
Cell	A particular area of space in a discretised 2D map, usually indicated by an x and y index pair or coordinate. The same as a cell in a spreadsheet.
Climbing Robot	A robot designed to be able to travel along vertical surfaces in some fashion.
Configuration-space	A mathematically abstract space whose dimensions are the parameters available for change by the robot. For example, a robot that has 2 joints would have a configuration space that is 2D. The axes of the space are the permissible values of that parameter.
End Effector	The device at the end of a robot manipulator which with the robot interacts with the environment.
Environment	The area or volume of space that the robot will operate in.
Ferromagnetic	Made of metals to which magnets are attracted.
Footpad	A link on the end of the robot that affixes itself to the surface.

Freespace	Space that is known to be empty of obstacles.
Frontier	The boundary between known free-space and unknown space.
Greedy	An algorithm in computing is considered greedy if it makes decisions based on short time-horizons, including as little as a single time-step.
Holonomic	An object (in the context of this thesis, a robot) is holonomic if its kinematic properties would allow it to freely move in any direction from a given position.
Information	A measure of uncertainty in the state of an environment/model.
Joint Effort	The sum of the difference in joint angles between two or more poses.
K-d Tree	A tree for storing points in k-dimensional space.
Kinematic Chain	A configuration of rigid bodies connected by joints.
Kinematics	The description of the motion of a system of objects.
Known Space	Space that is either known to be empty of obstacles, or known to contain an obstacle.
Manipulator Robot	A series of motors connected by joints, usually with one end fixed in place and the other end affixed with a tool or gripper.
Map	A representation of the environment, typically updated, maintained and used by the robot.
Occupancy Map	A discretised grid representation of the environment, made up of cells that store values representing the probability that that position in space contains an obstacle.
Octree	A compact method for storing a 3D occupancy map. The grid is stored as a tree so that when all eight children of a parent node have the same value, they can be deleted and the parent holds the value instead.
Obstacle	An impassable object.
Operator	A person in charge of controlling or monitoring a robot as it performs a task.
Point Robot	A robot that can be abstracted as a point in space.

Pose	The position and configuration of an object such as a robot, robot joint, end effector or sensor.
Quadtree	A compact method for storing a 2D occupancy map. The grid is stored as a tree so that when all four children of a parent node have the same value, they can be deleted and the parent holds the value instead.
Ray-cast/Ray-trace	Techniques for calculating the path of a ray through a grid or set of objects to determine which it interacts with.
Robot	A machine able to perform actions based on information it collects about the environment around it.
Surface Map	A map of 3D surface information.
Teleoperation	Operation of a machine from a distance by a human operator.
Triangle Mesh	A method for representing surfaces as a set of triangles connected by their edges and vertices.
Tether	A cable connecting the robot to either a control station, a power source, or a larger structure for safety, or a combination of these.
Unknown Space	Space that whose obstacle status is not known. It might or might not contain an obstacle.
Viewpoint	The position and orientation of a sensor in space, often as a result of a particular robot configuration, which determine what portion of the environment is covered by the sensor field of view.
Voxel	A particular volume of space in a 3D discretised map, usually represented by an (x,y,z) coordinate.